Lontar Komputer
JURNAL ILMIAH TEKNOLOGI INFORMASI

CURRENT ISSUE

Vol 14 No 2 (2023): Vol. 14, No. 2 August 2023

DOI: https://doi.org/10.24843/LKJITI.2023.v14.i02

PUBLISHED: 2023-11-06

ARTICLES

Detecting Pests and Diseases in Plants Using Efficient Network

Mardhiya Hayaty, Timur Haryo Mahissanular 71-79

PDF

Submit Articles Online

INFORMATION JOURNAL
Formatting Template
Publication Ethics
Peer Review Process
Editorial Team
For Authors
For Readers
For Librarians
Apply Reviewer

ACCREDITATION JOURNAL
DIKTI ACCREDITATION NO. 158/E/KPT/2021

UNUD JOURNAL RANKING NO. B/146/UN14.I/PJ.00.02/2022

VISITOR STATISTICS

Matomo

BROWSE

DOI : https://doi.org/10.24843/LKJITI.2023.v14.i02.p01
Abstract views: 45, PDF downloads: 30

Comparing Support Vector Machine and Naïve Bayes Methods with A Selection of Fast Correlation Based Filter Features in Detecting Parkinson's Disease

Yuniar Farida, Nurissaidah Ulinnuha, Silvia Kartika Sari, Latifatun Nadya Desinaini 80-90

PDF

DOI : https://doi.org/10.24843/LKJITI.2023.v14.i02.p02
Abstract views: 86, PDF downloads: 32

Digital Transformation Of Subak Management In Bali Through GIS Implementation

Shilta Inda Qurroti A'yun Achmadi, Anjela Faye M. Basco, Oka Sudana, Ni Kadek Dwi Rusjayanthi 91-101

PDF

DOI : https://doi.org/10.24843/LKJITI.2023.v14.i02.p03
Abstract views: 66, PDF downloads: 38

Refining Content-Based Segmentation for Prediction of Coffee Bean Quality

Suhendro Yusuf Irianto 102-113

PDF

DOI : https://doi.org/10.24843/LKJITI.2023.v14.i02.p04
Abstract views: 44, PDF downloads: 16

The Optimization of the ARP Poisoning Attack Detection Model Using a Similar Approach Based on NetFlow Analysis

Yohanes Priyo Atmojo, Dandy Pramana Hostiadi, I Made Darma Susila, Made Liandana, Gede Angga Pradipta, Putu Desiana Wulaning Ayu 114-125

PDF

DOI : https://doi.org/10.24843/LKJITI.2023.v14.i02.p05
Abstract views: 49, PDF downloads: 29

VIEW ALL ISSUES >

Lontar Komputer [ISSN Print 2088-1541] [ISSN Online 2541-5832] is a journal that focuses on the theory, practice, and methodology of all aspects of technology in the field of computer science and engineering as well as productive and innovative ideas related to new technology and information systems. This journal covers research original paper that has not been published and has been through the double-blind reviewed journal. Lontar Komputer is published three times a year by Research institutions and community service, University of Udayana. Lontar Komputer already indexing in **SINTA with score S2 and impact factor 4.28125.**

Lontar Komputer has been ACCREDITED by DIKTI (Ministry of Education, Culture, Research, and Technology) with a score SINTA 2, decree no. NO. 158/E/KPT/2021. DOI : 10.24843/LKJITI

By Author
By Issue

INDEXING

MEMBER OF

TOOL

CURRENT ISSUE

SERTIFIKAT

Direktorat Jenderal Pendidikan Tinggi, Riset dan Teknologi
Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi Republik Indonesia

Kutipan dari Keputusan Direktorat Jenderal Pendidikan Tinggi, Riset dan Teknologi
Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia

Nomor 158/E/KPT/2021
Peringkat Akreditasi Jurnal Ilmiah Periode I Tahun 2021

Nama Jurnal Ilmiah

Lontar Komputer: Jurnal Ilmiah Teknologi Informasi

E-ISSN: 25415832

Penerbit: Lembaga Penelitian dan Pengabdian Kepada Masyarakat Universitas Udayana

Ditetapkan Sebagai Jurnal Ilmiah

TERAKREDITASI PERINGKAT 2

Akreditasi Berlaku selama 5 (lima) Tahun, yaitu
Volume 12 Nomor 1 Tahun 2021 Sampai Volume 16 Nomor 2 Tahun 2025

Jakarta, 09 December 2021
Plt. Direktur Jenderal Pendidikan Tinggi,
Riset, dan Teknologi

Prof. Ir. Nizam, M.Sc., DIC, Ph.D., IPU, ASEAN Eng
NIP. 196107061987101001

LONTAR KOMPUTER VOL. 14, NO. 2 AUGUST 2023
DOI : 10.24843/LKJITI.2023.v14.i02.p05
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

p-ISSN 2088-1541
e-ISSN 2541-5832

The Optimization of the ARP Poisoning Attack Detection Model Using a Similar Approach Based on NetFlow Analysis

Yohanes Priyo Atmojo[a1], Dandy Pramana Hostiadi[b2], I Made Darma Susila[a3], I Made Liandana[a4], Gede Angga Pradipta[b5], Putu Desiana Wulaning Ayu[b6]

[a]Department Information Technology, Faculty of Informatics and Computer, Institut Teknologi dan Bisnis STIKOM Bali
Jl. Raya Puputan, Denpasar-Bali, Indonesia
[1]yohanes@stikom-bali.ac.id
[3]darma_s@stikom-bali.ac.id[3]
[4]liandana@stikom-bali.ac.id

[b]Department of Magister Information Systems, Institut Teknologi dan Bisnis STIKOM Bali
Jl. Raya Puputan, Denpasar-Bali, Indonesia
[2]dandy@stikom-bali.ac.id
[5]angga_pradipta@stikom-bali.ac.id
[6]wulaning_ayu@stikom-bali.ac.id

Abstract

Information security and threats are a concern in the cyber era. Attacks can be malicious activities. One of them is known as ARP poisoning attack activity, which attacks by falsifying a computer's identity through illegal access to retrieve confidential information in a target computer. Besides, it has also caused service deadlocks in the network. Previous studies have been introduced for the ARP Attack Detection model using rule-based and mining-based. Still, they cannot show optimal detection performance and obtain high false positive results. This paper proposed a detection model for ARP poisoning attacks using a similarity measurement approach adopting cosine similarity. The goal is to obtain measurements of host activities similar to ARP poisoning attacks. The experiment results showed that the model got an accuracy of 97.25%, recall of 96.43%, and precision of 81% with a similarity threshold value of 0.488. Comparison results with previous studies showed higher detection accuracy than previous studies and some classification methods. It shows that the model can improve intrusion detection performance and facilitate network administrators to analyze ARP poisoning attacks in computer networks.

Keywords: *ARP Poisoning, Similarity, Machine Learning, Network Security, Network Infrastructure*

1. Introduction

Nowadays, system security in computer networks needs to be handled appropriately. It also needs to avoid and detect malicious activities that can cause worse damage to computer networks and credential data. This malicious activity is often referred to as intrusion [1]. In dealing with intrusion in computer networks, an Intrusion detection model known as IDS [2] needs to be implemented in computer networks to strengthen communication services between interconnected computers.

Generally, IDS systems are built with two approaches, namely misuse detection and anomaly detection [3]. These approaches can be implemented based on rule-based formation, such as the SNORT application [4]–[7]. However, this IDS model cannot detect new types of attack variants in the network, and the detection accuracy depends on the accuracy of the rule base formation.

One type of attack with a high detection error in intrusion detection systems is the Address Resolution Protocol (ARP) Poisoning attack. ARP is a communications protocol that maps the addressing of each computing device in the computer network as a Media Access Control (MAC) address. Research on ARP attack detection has been introduced in previous research, such as detection models by applying the concept of mining-based analysis [1]–[3], [8], [9], anomaly behavior analysis [10]–[13] and used supporting applications [4], [14], [15]. In [16], an ARP poisoning detection model is introduced by separating ARP service access and using a listing technique in the form of static MAC address addressing. The experimental results show the model can detect suspected ARP Poisoning attack activity. Still, the model is very dependent on resources in the ARP control flow model. Selvarajan et al. [5] proposed an ARP poisoning detection model involving communication analysis on manipulated ICMP echo requests. The detection process involves rule-based analysis by checking communications on the ICMP protocol on ping messages that lead to ARP spoofing activity. The test results show that the model can detect behavior using a statistical approach based on the behavioral characteristics of suspected ARP poisoning perpetrators who communicate using certain protocol services. However, this research requires an attack scenario using a specific protocol. This type of ARP poisoning attack generally involves several communication protocols. [13] proposed an ARP attack detection model by distinguishing detection and prevention mechanisms. ARP attacks are divided into two activity processes: ARP spoofing and ARP poisoning attacks. The activity detection mechanism creates a static list of communications in the network, namely a list of IP address mapping and MAC address mapping tables. Then, the legal access in the permission list gets communication registration on the network. This model produces the detection and prevention of ARP poisoning attacks but requires a validation process on the network, which can hinder the duration of activity on the network. Besides, the model introduced uses statistical rules implemented for several hosts declared to be registered. In some cases, new hosts that have not yet been registered require a legal communication process.

The ARP poisoning model often uses analysis of network traffic data obtained from recordings [15] or uses public datasets that have been processed in the form of netflows [8], [10]. Besides, the Netflows are commonly used to detect attacks or malicious activity anomalies [17], including ARP Poisoning attacks. ARP poisoning is often misused to cause computer network deadlocks by illegally forging computer addresses to obtain confidential and credential information [8]. Thus, the proper ARP poisoning attack detection technique is required to accurately detect the attack in a computer network.

This paper proposed a new approach for ARP poisoning attack detection using a similarity analysis based on NetFlow analysis. This research is a development in [8], where the detection model used a classification approach in previous studies. The results of ARP poisoning attack detection using classification have a high detection accuracy but a high detection error value. Therefore, in the proposed research, the detection model aims to improve detection accuracy by suppressing detection errors based on NetFlow analysis. The novelty of this research is that it involves analyzing the dynamic similarity threshold value. In addition, this research built a knowledge base based on the characteristics of attack patterns. The similarity measurement results in the detection model are expected to show the closeness between the activity patterns of suspected attackers and the characteristics of the ARP poisoning attack knowledge base.

This paper is constructed into several sections. The process stages of the proposed model are introduced in Section II. Section III presents the results of the experiment and the research discussion. Finally, the conclusions of the research are drawn in section IV.

2. Research Methods

Research on ARP poisoning attack detection has been conducted by previous researchers. Some of them applied the concept of mining-based analysis [1]–[3], [8], [9], anomaly behavior analysis [10]–[13] and used supporting applications [4], [14], [15]. ARP poisoning activities often used analysis on network traffic data obtained from recording results [15] or used public datasets that had been processed in the form of network traffic flows (netflows) [8], [10]. The proposed ARP Poisoning detection model is shown in Figure 1.

p-ISSN 2088-1541
e-ISSN 2541-5832

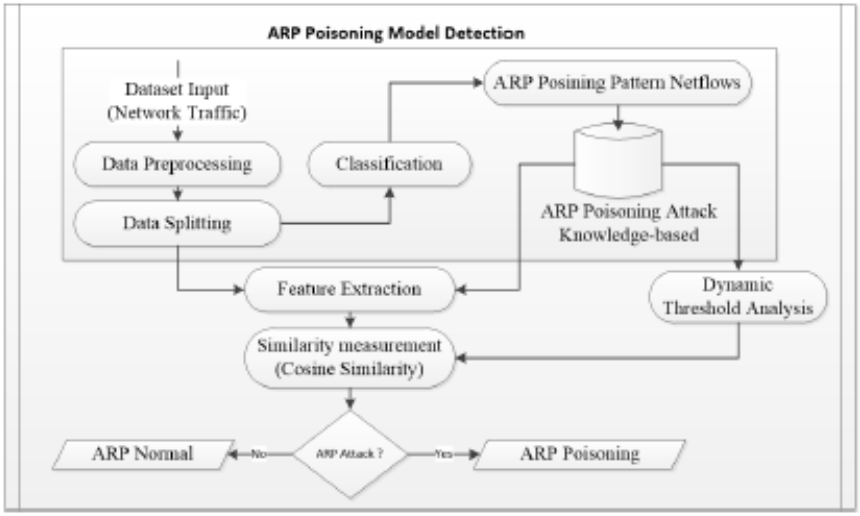

Figure 1. The proposed model

2.1. Dataset

This research used network traffic data obtained from the computer network recording process. The recorded data was in the form of .pcap files and processed into .csv files to form a dataset by adopting the techniques in research [8] and standardized based on the IDMEF standard [18]. Netflow data had a recording duration of 1 hour with 2819 traffic records, 279 ARP poisoning attack records, and 2540 normal activity traffic records. The number of attackers was six out of 418 hosts. ARP poisoning attacks involved malware to do ARP broadcasts, ARP flooding packets, and MAC flooding. Normal activities were conducted by hosts with browsing, e-mail sending, DNS access, and FTP access.

2.2. Data Preprocessing

At this phase, the data in .csv format are processed in the preprocessing process: data cleansing, normalization, and feature selection. The data cleansing stage was conducted to clean some data records that do not have values and are double data. Double data can appear due to errors in the recording process. In addition, the data cleansing process was carried out to fill in the "0" value to the feature value with a null value, aiming to standardize the value of each attribute in the data record. After the cleansing process, the normalized data were normalized using a value range of 0 to one, adopting the approach in research [19].

2.3. Data Splitting

Data splitting is dividing data into two types: training data and testing data. This stage was often used for the data learning process of a machine learning-based detection model [20]–[23] with a composition of 70% and 30% as testing data. In the testing data, the attack identities label of each record is removed and used to test the learning model used.

2.4. Classification

The classification stage is the ARP attack activity detection stage from the learning process. The detection results would record the attack activity and be stored in the knowledge base. The classification stage used five types of classification models, namely k-NN, Logistic Regression, Naïve Bayes, Random Forest, and Decision Tree. Evaluation of the five models with the best detection accuracy, precision, and recall values stored would be labeled as ARP poisoning pattern NetFlow.

2.5. ARP Poisoning Pattern NetFlow

ARP poisoning pattern flows are the knowledge base of ARP attack activities. Each data record would be labeled as an activity and sorted based on the time of the ARP attack activity. Thus, in the knowledge base of sequential activities among ARP attack activities.

2.6. Dynamic Threshold Analysis

Dynamic threshold analysis is a stage to determine the relationship between ARP attack activities. The intended relationship was the similarity between the two attacker activities. If two different attackers are represented as nodes A and B, there may be a difference in similarity between A to B and B to A [24]. To determine whether two attacking objects are similar and have a substantial similarity value, it is necessary to analyze the similarity threshold value [25]. In this paper, the similarity measurement (τ_{sim}) between two ARP attack patterns adopted the cosine similarity approach. The determined threshold value was dynamic based on the characteristics of the data using the (1):

$$T_{sim} = \frac{Max_{sim} + Min_{sim}}{2}, \quad (1)$$

where Min_{sim} Is the lowest similarity value that occurs from all the attacker activity similarity measurements, Max_{sim} is the highest value of the measured similarity value. The threshold value obtained will be updated if new attack characteristics are in the knowledge base update, namely in the ARP poisoning pattern NetFlow.

2.7. Feature Extraction

The Feature Extraction stage is the feature extraction of each data attribute or primary feature in NetFlow traffic. The number of primary attributes used in the previous process was ten features: *source port, source IP address, destination port, destination IP address, length, UDP port, TCP port, source MAC address, protocol, and destination MAC address.* If traffic in a computer network (N_T) consists of host activities in which there are traffic records (r_t) defined as $R_t = \{r_{t_1}, r_{t_2}, \dots, r_{t_j}\}$, containing feature *tuples* (F) in each record, namely *source IP address* (F_{SIP}), *destination IP address* (F_{DIP}), *protocol*(F_{Proto}), *TCP port* (F_{TPort}), *UDP port*(F_{UPort}), *length*(F_{length}), *source port*(F_{SPT}), *source MAC address*(F_{MSrc}) and *destination MAC address* (F_{Mdst}), thus denoted $n_T \in N_T$, where $n_T = \{(F_{SIP}, F_{DIP}, F_{TPort}, F_{UPort}, F_{length}, F_{SPT}, F_{Proto}, F_{MSrc}, F_{Mdst},)\}$.

Each feature tuple (F) value was re-extracted to get six new features denoted as FT by calculating the activity type of each host's interaction. The type of host activities can be in the form of spreading activity and show their pattern for communication behavior in the network. The type or variant of the feature is calculated by grouping each feature based on (F_{SIP}). Thus, each feature is defined as$FT_1 = n(F_{TPort}), FT_2 = n(F_{UPort}), FT_3 = n(F_{length}), FT_4 = n(F_{SPT}), FT_5 = n(F_{DIP}), FT_6 = n(F_{Proto})$.

The feature extraction results produced a feature pattern denoted as φ with a value of $\varphi = \{FT_1, FT_2, FT_3, FT_4, FT_5, FT_6,\}$. The feature pattern ($\varphi$) obtained from the classification results will be stored in the ARP poisoning attack activity knowledge base denoted by φ_{kb}.

2.8. Similarity Measurement

This stage measured the similarity between ARP attack traffic in the knowledge base as training data and network traffic as testing data. The similarity measurement adopted the cosine similarity shown in (2).

$$sim_{cos}(x,y) = \cos(x,y) = \frac{x.y}{|x||y|}, \quad (2)$$

where the inner product is symbolized by the sign "." with the calculation:

$$x.y = \sum_{i=1}^{r} x_i y_i \quad (3)$$

and $|x|$ the calculation result of vector x :

$$|x| = \sqrt{\sum_{i=1}^{r} x_i^2} \quad (4)$$

and $|y|$ the calculation result of vector y :

$$|y| = \sqrt{\sum_{i=1}^{r} y_i^2} \quad (5)$$

If the traffic testing data is denoted as dt_ts, then the feature tuple extraction on the testing data becomes $F - dt_ts$, with the features $(FT - dt_ts)$ formed into a feature pattern denoted as φ_{dt_ts}. Thus, the similarity measurement to determine how substantial the similarity is with the cosine similarity between the ARP poisoning attack feature pattern in the knowledge base (φ_{kb}) and the traffic in the testing data (φ_{dt_ts}) become (6).

$$\text{sim}\omega_{(\varphi_{kb},\varphi_{dt_ts})} = \frac{\sum_{i=1}^{r} \varphi kb_i \varphi dt_ts_i}{|\sqrt{\sum_{i=1}^{r} \varphi kb_i^2}||\sqrt{\sum_{i=1}^{r} \varphi dt_ts_i^2}|}, \quad (6)$$

The similarity measurement results $sim\omega_{(\varphi_{kb},\varphi_{dt_ts})}$ Which has a value above τ_{sim} will state φ_{dt_ts} as ARP poisoning attack activity. Thus, it is expressed in (7):

$$sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} = \begin{cases} \text{state ARP Poisoning Attack if } sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} \geq \tau_{sim} \\ \text{state Normal aktivity if } sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} < \tau_{sim} \end{cases}, \quad (7)$$

2.9. Evaluation

In this stage, the evaluation uses F-measure by measuring the precision, recall, and detection accuracy. To calculate the True Positive, True Negative, False Negative, and False Positive values are traced. The true Negative (TN) value is the number of regular activities detected as normal activities. False Positive (FP) is a normal activity but detected as an ARP poisoning attack activity. Meanwhile, True Positive (TP) is an ARP poisoning attack activity that is correctly detected as an ARP poisoning attack activity. False Negative (FN) is the opposite of True Positive, so ARP poisoning attack activity is detected as normal activity.

3. Result and Discussion

This research used the ARP poisoning attack dataset used in the study [8]. Network traffic data was taken through network traffic recording with the assistance of the Wireshark application [26] and produced files in the form of .pcap. The recording was carried out for 1 hour. This research used a computer with Intel Core i5-9300H processor specifications, 16 GB RAM, and 500 GB SSD storage capacity during the experiment and traffic data collection on the network.

3.1. Experiment

The recording data produced by the application was the .pcap extension. This data was processed into a comma-separated value (.csv) file. It aimed to convert traffic data from unstructured to structured tabular data by separating each column with a ";" separator. The conversion was done using the command line-based Tshark application, as shown in Figure 2.

p-ISSN 2088-1541
e-ISSN 2541-5832

```
root@ubuntu:/home/sysadmin/arp# tshark -r TCPdump_Inside_Labeling.pcap -T fields -e frame.time -e ip
.src -e eth.src -e tcp.port -e udp.port -e frame.len -e ip.proto -e ip.dst -e eth.dst -E separator=,
 -E occurrence=f
```

Figure 2. Tshark command

After that, preprocessing was carried out, namely data cleansing, data normalization, and feature selection. In the data cleansing stage, 153 traffic records were deleted because they were redundant data. Redundant data could occur during the recording to conversion process using two applications, Wireshark and Tshark. In addition to removing redundant data, the null value in each column was filled to 0. There were 201 record data that had columns with null values and converted to a 0 value. The data cleansing process resulted in a traffic reduction of 5.43%. The details of the cleansing results are shown in Table 1.

Table 1. Preprocessing Result

Data before preprocessing	Data after preprocessing	Reduction Percentage
2,819	2,666	5.43 %

The results of data cleansing were followed by converting the values in each category data column into numerical values. This change technique used an encoder, also done in research [27]. Furthermore, all values in each column were normalized to a value scale between 0 and 1. Each normalized column was expressed as a traffic data feature. The basic features obtained with the Tshark application in Figure 2 of 7 features. In this research, six features were selected, and the selection was conducted manually. One feature that was not used was the time feature. The time feature was not used because it did not match the characteristics of ARP poisoning attacks, where attack activities occurred randomly and not continuously. ARP attacks tend to be characterized by spreading and have high-intensity occurrences. Features used were *source IP* (F_{SIP}), *destination IP* (F_{DIP}), *TCP port* (F_{TPort}), *UDP port*(F_{UPort}), *length*(F_{length}), *source port*(F_{SPT}), *protocol*(F_{Proto}), *source MAC address*(F_{MSrc}) and *destination MAC address* (F_{Mdst}).

From the preprocessing results, the data was divided into two types of data, namely training data as the knowledge base of ARP attacks (φ_{kb}) after the classification stage and testing data as denoted by (φ_{dt_ts}). Data were divided randomly with data divided composition shown in Table 2.

Table 2. Data Composition

Traffic Record		Data Training (70%)		Data Testing (30%)	
ARP attack	Normal	ARP attack	Normal	ARP attack	Normal
279	2,387	195	1,671	84	716
Total Traffic after Preprocessing: 2,666		Total Traffic: 1,866		Total Traffic: 800	

In this research, the classification was performed using five classification methods, namely *k*-NN, Logistic Regression, Naïve Bayes, Random Forest, and Decision Tree. The classification results are shown in Figure 3.

p-ISSN 2088-1541
e-ISSN 2541-5832

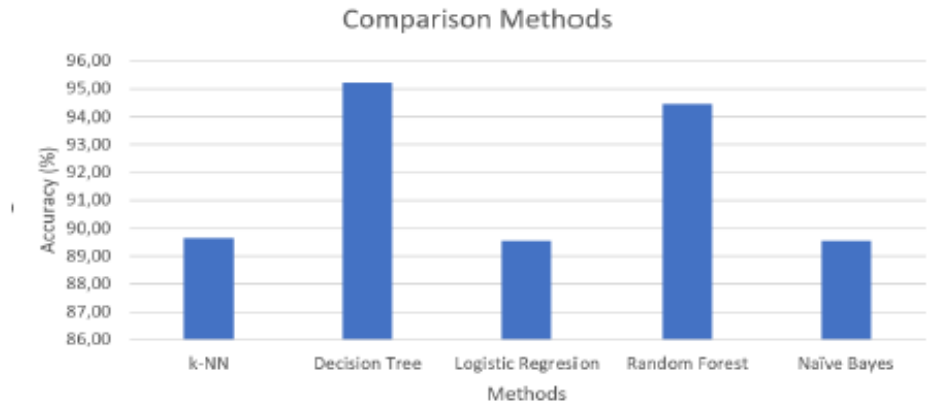

Figure 3. Evaluation results of 5 classification methods

The method with the best detection accuracy will be used to build the ARP poisoning attack knowledge base. The results of the five classification methods are shown in Table 3.

Table 3. Classification result

No.	Classification method	Accuracy
1	k-NN	89.65
2	Random Forest	94.45
3	Naïve Bayes	89.53
4	Logistic Regression	89.53
5	Decision Tree	95.24

The classification results showed that the decision tree method obtained the highest detection accuracy by producing 196 traffic records as ARP poisoning attack patterns. Furthermore, the traffic records were extracted into FT, where the variants of each feature were grouped based on (F_{SIP}) and produced a total of $n = (FT_i)$ with i = 6 namely $F_{TPort}, F_{UPort}, F_{length}, F_{SPT}, F_{DIP}, F_{Proto}$. The feature extraction results formed feature patterns $\varphi = \{FT_1, FT_2, FT_3, FT_4, FT_5, FT_6,\}$ with the value of each FT is $FT_1 = n(F_{TPort}), FT_2 = n(F_{UPort}), FT_3 = n(F_{length}), FT_4 = n(F_{SPT}), FT_5 = n(F_{DIP}), FT_6 = n(F_{Proto})$. Examples of the knowledge base (φ_{kb}) of ARP poisoning attack patterns is shown in Table 4.

Table 4. Traffic records on φ_{kb}

φ_{kb_i}	$n(F_{TPort})$	$n(F_{UPort})$	$n(F_{length})$	$n(F_{SPT})$	$n(F_{DIP})$	$n(F_{Proto})$	ARP Poisoning Attack
1	0.7727	0.2857	0.8421	0.3333	0.4795	0.9091	1
2	0.1364	0.4762	0.9474	0.8182	0.6438	0.9091	1
3	0.6364	0.7143	0.8947	0.3939	0.3562	0.9091	1
4	0.6136	0.1429	0.1053	0.3030	0.7808	0.9091	1
5	0.8636	0.1905	0.5789	0.9697	0.6712	0.9091	1
6	0.4318	0.8571	0.1053	0.5455	0.0548	0.9091	1
7	0.4318	0.6190	0.1579	1,0000	0.0959	0.9091	1
8	0.3864	0.3810	0.3684	0.7879	0.0685	0.9091	1
9	0.4318	0.4286	0.1579	0.6970	0.9589	0.9091	1
10	0.6136	0.2857	0.3684	0.6970	0.9863	0.9091	1
11	0.4091	0.3810	0.5789	0.7879	0.4521	0.9091	1
12	0.9545	0.3333	0.0526	0.2424	0.2329	0.9091	1
13	0.7500	0.4286	0.1579	0.3030	0.6301	0.9091	1
14	0.0455	0.5714	0.8947	0.5455	0.0137	0.8182	1
15	0.5455	0.8571	0.7895	0.0303	0.3836	0.8182	1
...	...	...	...	...	...	...	...

In the similarity measurement, the testing data used was 30% of the dataset divided at the data splitting stage. Examples of testing data that have been formed into feature patterns φ_{dt_ts} are shown in Table 5.

Table 5. Data Testing

$\varphi_{dt_ts_i}$	$n(F_{TPort})$	$n(F_{UPort})$	$n(F_{length})$	$n(F_{SPT})$	$n(F_{DIP})$	$n(F_{Proto})$	ARP Poisoning Attack
1	0.2045	0.1429	0.0526	0.5455	0.0137	0.2727	?
2	0.2045	0.2381	0.0000	0.6364	0.5890	0.8182	?
3	0.5682	0.7619	0.1579	0.8788	0.6301	0.8182	?
4	0.3864	0.1905	0.8947	0.3030	0.6712	0.2727	?
5	0.2727	0.3810	1,0000	0.6364	0.1096	0.0909	?
6	0.4545	0.6667	0.7895	0.0606	0.7808	0.4545	?
7	0.2045	0.2857	0.7368	0.8788	0.1781	0.8182	?
8	0.4318	0.6667	0.6842	1,0000	0.6301	0.5455	?
9	0.2727	0.4286	0.5789	0.6970	0.8082	0.5455	?
10	0.3409	0.9524	0.4737	0.7576	0.4658	0.2727	?
11	0.9545	0.6190	1,0000	0.2121	0.5205	0.3636	?
...	...	...	...	...	...	...	...

The results of similarity measurement between the ARP poisoning attack feature pattern in the knowledge base (φ_{kb}) and the traffic in the testing data (φ_{dt_ts}) result in several similarity values. Each data record in φ_{dt_ts} will be measured against 196 data records in φ_{kb}. The value taken as the similarity result was the average value of the total similarity with the φ_{kb}. Data record. The results of similarity measurement are shown in Table 6.

Table 6. Example of Cosine Matrix Map Results

		φ_{dt_ts}										
		1	2	3	4	5	6	7	8	9	10	11
φ_k	1		0.197	0.141	0.519	0.368	0.585	0.146	0.165	0.270	0.246	...
	2	0.197		0.075	0.400	0.563	0.371	0.200	0.175	0.132	0.288	...
	3	0.141	0.075		0.327	0.388	0.249	0.176	0.071	0.097	0.100	...
	4	0.519	0.400	0.327		0.160	0.086	0.243	0.158	0.109	0.265	...
	5	0.368	0.563	0.388	0.160		0.282	0.171	0.155	0.235	0.202	...
	6	0.585	0.371	0.249	0.086	0.282		0.325	0.181	0.133	0.193	...
	7	0.146	0.200	0.176	0.243	0.171	0.325		0.092	0.133	0.218	...
	8	0.165	0.175	0.071	0.158	0.155	0.181	0.092		0.033	0.051	...
	9	0.270	0.132	0.097	0.109	0.235	0.133	0.133	0.033		0.117	...
	10	0.246	0.288	0.100	0.265	0.202	0.193	0.218	0.051	0.117		...
	...	...	...	...	...	...	...	...	...	...	...	...

From the similarity measurement results, the lowest value of $sim\omega_{(\varphi_{kb},\varphi_{dt_ts})}$ was 0.033, and the highest value was 0.862. Thus, the value of τ_{sim} obtained was 0.448 as the threshold value for determining ARP poisoning traffic. In this research, the value of τ_{sim} = 0.448 successfully obtained six hosts of ARP Poisoning attack perpetrators based on the *source IP* (F_{SIP}) grouping with the identification accuracy of the number of ARP poisoning attack traffic 82 and normal activity of 709 records. Identification details of ARP poisoning attacks based on similarity measurements are shown in Table 7.

Table 7. Detailed evaluation results of ARP Attack detection

TP	FP	FN	TN	Accuracy	Precision	Recall
82	7	2	709	98.88	92.13	97.62

p-ISSN 2088-1541
e-ISSN 2541-5832

The similarity measurement results successfully detect ARP Poisoning traffic records with a detection accuracy of 98.88%, precision of 92.13%, and recall of 97.62%. It showed that the proposed model performs optimally to detect ARP poisoning attacks.

3.2. Analysis and Discussion

This research proposed a new ARP Poisoning attack detection model with a similarity measurement approach and dynamic threshold value analysis. In the preprocessing stage, there was a traffic reduction of 5.43%, and only normal activity occurred. Some normal activities had the potential to be performed repeatedly by the user, thus causing recording as redundant data by the Wireshark application and Tshark application conversion.

In analyzing threshold value, in addition to using (1), this research also conducted a heuristic analysis to obtain the optimal τ_{sim} value analysis. The result of the similarity threshold value search found that the best τ_{sim} value was 0.488, with the highest accuracy, precision, and recall values. Thus, it was determined that τ_{sim} =0.488 was the optimal threshold value used to detect ARP Poisoning attacks on traffic records in the testing data. The threshold value can dynamically change if there is a change in the results of the knowledge base formation that depends on the classification method. The higher the accuracy value of the classification method, the more optimal the knowledge base formation in the form of ARP attack feature patterns. The optimization of knowledge base formation shows a more significant amount of feature pattern data. The results of the threshold value search are shown in Figure 4.

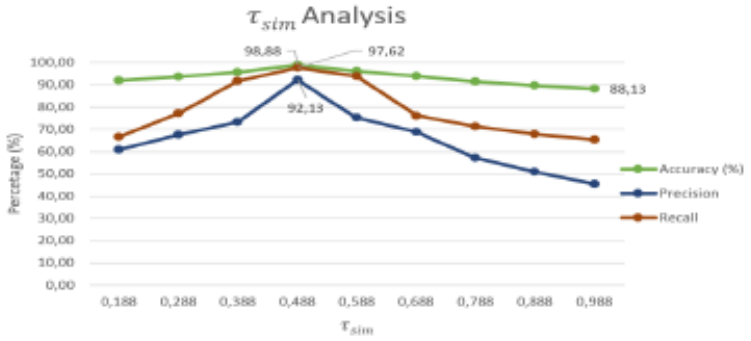

Figure 4. Optimal value tracing results from τ_{sim}

In this paper, the proposed model could detect ARP Poisoning attacks with a detection accuracy value of 97.25%, precision of 81%, and recall of 96.43%. These results had higher values than research [8] and detection results in several classification methods. The comparison results are shown in Table 8.

Table 8. Comparison with previous studies

Methods and Previous Studies	Accuracy	Precision	Recall	Threshold analysis
Atmojo et al. [8] k-NN (k=2)	98,7	98,7	98,6	No
Random Forest	94,5	95,1	99,2	No
Linear Regression	91,9	92,1	99,6	No
Naïve Bayes	96,7	97,7	99	No
Support Vector Machine	95,6	96,8	98,8	No
Decision Tree	97,5	99,0	98,5	No
Proposed Model	**98.88**	**92.13**	**97.62**	**yes**

The comparison results with previous research; the proposed model had higher detection accuracy than earlier research, which was 98.88%. The detection accuracy that could be obtained in this research was 0.1% higher than in the previous study. However, the precision and recall values had lower values. The precision value was 6.87% lower than the highest precision value produced by the Decision Tree method. At the same time, the recall value was 1.92% lower than the highest recall value produced by the Linear Regression classification method. The precision and recall values were obtained lower because the composition of the number of ARP attack activities was only 10.5% of the total traffic data in the testing data and imbalanced compared to the number of normal activity data records. Besides, this paper has the novelty of a detection process that involves the analysis of dynamic similarity threshold values that have never been used in previous research.

4. Conclusion

This paper proposed a detection model to detect ARP poisoning attacks using a similarity analysis approach adopting cosine similarity. The proposed model aimed to obtain substantial similarity between host activities in the network and ARP Poisoning attack activities in the knowledge base formed from the machine learning-based classification model. The proposed model had a novelty in analysis that involved a dynamic threshold value to determine the host activity pattern as an ARP Poisoning attack. In this research, the model successfully detected ARP Poisoning attacks with a detection accuracy value of 97.25%, precision of 81%, and recall of 96.43% with a similarity threshold value of 0.488. Detection accuracy showed higher results than Atmojo et al. [8] and some machine learning-based classification methods. The detection accuracy that could be obtained in this research was 0.1% higher than the highest accuracy value in previous research obtained by the k-NN classification method with a value of $k = 2$. However, the recall and precision had lower values. The precision value was 6.87% lower than the highest value the Decision Tree method produced. At the same time, the recall value was 1.92% lower than the highest recall value produced by the Linear Regression classification method. These two lower precision and recall values were caused by the data composition formed during network traffic recording. However, the composition of the number of traffic records recorded with the Wireshark application was a characteristic that corresponded to the actual occurrence of ARP Poisoning attacks on the network. In this paper, the proposed model could be used to develop intrusion detection models and make it easier for network administrators to analyze ARP poisoning attacks in computer networks.

The model needs to be developed in future research by optimizing features and feature selection. It aims to improve the model evaluation of the precision and recall measurements without degrading the current detection accuracy results. The precision and recall values need to be increased in further research by handling the imbalance data issues to reduce the positive error rate. In addition, it can develop a time of occurrence analysis to obtain attack analysis that can be causal attacks. Thus, the ARP Poisoning attack detection model could be performed optimally.

References

[1] G. Pu, L. Wang, J. Shen, and F. Dong, "A hybrid unsupervised clustering-based anomaly detection method," *Tsinghua Tsinghua Science and Technology*, vol. 26, no. 2, pp. 146–153, 2021, doi: 10.26599/TST.2019.9010051.

[2] V. Bhatia, S. Choudhary, and K. R. Ramkumar, "A Comparative Study on Various Intrusion Detection Techniques Using Machine Learning and Neural Network," *ICRITO 2020 - IEEE 8th Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, pp. 232–236, 2020, doi: 10.1109/ICRITO48877.2020.9198008.

[3] L. Mohan, S. Jain, P. Suyal, and A. Kumar, "Data mining Classification Techniques for Intrusion Detection System," *Proceedings - 2020 12th International Conference on Computational Intelligence and Communication Networks, CICN 2020*, pp. 351–355, 2020, doi: 10.1109/CICN49253.2020.9242642.

[4] A. Tasneem, A. Kumar, and S. Sharma, "Intrusion Detection Prevention System using SNORT," *International Journal of Computer Applications*, vol. 181, no. 32, pp. 21–24, 2018, doi: 10.5120/ijca2018918280.

[5] S. Selvarajan, M. Mohan, and B. R. Chandavarkar, "Techniques to Secure Address Resolution Protocol," *2020 11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020*, doi: 10.1109/ICCCNT49239.2020.9225413.

[6] Z. Trabelsi and W. El-Hajj, "ARP spoofing: A comparative study for education purposes," *Proceedings of the 2009 Information Security Curriculum Development Annual Conference, InfoSecCD'09*, pp. 60–66, 2009, doi: 10.1145/1940976.1940989.

[7] V. Srivastava and D. Singh, "Enhance detecting and preventing scheme for ARP Poisoning using DHCP," Computer Modelling and New Technologies, vol. 21, no. 2, pp. 93–99, 2017.

[8] Y. P. Atmojo, I. M. D. Susila, I. B. Suradarma, L. Yuningsih, E. S. Rini, and D. P. Hostiadi, "A New Approach for ARP Poisoning Attack Detection Based on Network Traffic Analysis," *2021 4th International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2021*, pp. 18–23, 2021, doi: 10.1109/ISRITI54043.2021.9702860.

[9] A. Krishna, M. A. Ashik Lal, A. J. Mathewkutty, D. S. Jacob, and M. Hari, "Intrusion Detection and Prevention System Using Deep Learning," *Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020*, pp. 273–278, 2020, doi: 10.1109/ICESC48915.2020.9155711.

[10] V. Rohatgi and S. Goyal, "A detailed survey for detection and mitigation techniques against ARP spoofing," Proceedings of the 4th International Conference on IoT in Social, Mobile, Analytics, and Cloud, ISMAC 2020, pp. 352–356, 2020, doi: 10.1109/I-SMAC49090.2020.9243604.

[11] M. Ren, Y. Tian, S. Kong, D. Zhou, and D. Li, "A detection algorithm for ARP man-in-the-middle attack based on data packet forwarding behavior characteristics," *Proceedings of 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference, ITOEC 2020*, pp. 1599–1604, 2020, doi: 10.1109/ITOEC49072.2020.9141555.

[12] S. Sun, X. Fu, B. Luo, and X. Du, "Detecting and mitigating ARP attacks in SDN-based cloud environment," *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops 2020*, pp. 659–664, 2020, doi: 10.1109/INFOCOMWKSHPS50562.2020.9162965.

[13] S. Hijazi and M. S. Obaidat, "A New Detection and Prevention System for ARP Attacks Using Static Entry," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2732–2738, 2019, doi: 10.1109/JSYST.2018.2880229.

[14] S. Ahn, T. Lee, and K. Kim, "A Study on Improving Security of ICS through Honeypot and ARP Spoofing," *ICTC 2019 - 10th International Conference on ICT Convergence: ICT Convergence Leading the Autonomous Future*, pp. 964–967, 2019, doi: 10.1109/ICTC46691.2019.8939925.

[15] M. Abid and A. Singh, "Arp Spoofing Detection via Wireshark and Veracode," *International Journal of New Technology and Research*, vol. 4, no. 5, p. 263063, 2018.

[16] H. Y. Ibrahim, P. M. Ismael, A. A. Albabawat, and A. B. Al-Khalil, "A Secure Mechanism to Prevent ARP Spoofing and ARP Broadcasting in SDN," *Proceedings of the 2020 International Conference on Computer Science and Software Engineering, CSASE 2020*, pp. 13–19, 2020, doi: 10.1109/CSASE48920.2020.9142092.

[17] T. Yu and R. Yue, "Detecting Abnormal Interactions among Intranet Groups Based on Netflow Data," *IOP Conference Series: Earth and Environmental Science*, vol. 428, no. 1, 2020, doi: 10.1088/1755-1315/428/1/012039.

[18] H. Debar, "The IDMEF : RFC 4765," *Mycological Research*, 2007.

[19] D. P. Hostiadi and T. Ahmad, "Hybrid model for bot group activity detection using similarity and correlation approaches based on network traffic flows analysis," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 7, pp. 4219–4232, 2022, doi: 10.1016/j.jksuci.2022.05.004.

[20] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *Journal Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-020-00390-x.

[21] M. N. Aziz and T. Ahmad, "Clustering under-sampling data for improving the performance of intrusion detection system," *Journal of Engineering Science and Technology*, vol. 16, no. 2, pp. 1342–1355, 2021.

[22] F. Selahshoor, H. Jazayeriy, and H. Omranpour, "Intrusion Detection systems using Real-Valued Negative Selection Algorithm with Optimized Detectors," *5th Iranian Conference on Signal Processing and Intelligent Systems, ICSPIS 2019*, pp. 18–19, 2019, doi: 10.1109/ICSPIS48872.2019.9066040.

The Optimization of the Poisoning Attack Detection Model Using a Similar Approach Based on NetFlow Analysis

By Angga Pradipta

WORD COUNT	6194	TIME SUBMITTED	29-NOV-2023 12:57PM
		PAPER ID	104686995

LONTAR KOMPUTER VOL. 14, NO. 2 AUGUST 2023
DOI : 10.24843/LKJITI.2023.v14.i02.p05
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021
p-ISSN 2088-1541
e-ISSN 2541-5832

The Optimization of the ARP Poisoning Attack Detection Model Using a Similar Approach Based on NetFlow Analysis

Yohanes Priyo Atmojo[a1], Dandy Pramana Hostiadi[b2], I Made Darma Susila[a3], I Made Liandana[a4], Gede Angga Pradipta[b5], Putu Desiana Wulaning Ayu[b6]

[a]Department Information Technology, Faculty of Informatics and Computer, Institut Teknologi dan Bisnis STIKOM Bali
Jl. Raya Puputan, Denpasar-Bali, Indonesia
[1]yohanes@stikom-bali.ac.id
[3]darma_s@stikom-bali.ac.id[3]
[4]liandana@stikom-bali.ac.id

[b]Department of Magister Information Systems, Institut Teknologi dan Bisnis STIKOM Bali
Jl. Raya Puputan, Denpasar-Bali, Indonesia
[2]dandy@stikom-bali.ac.id
[5]angga_pradipta@stikom-bali.ac.id
[6]wulaning_ayu@stikom-bali.ac.id

Abstract

Information security and threats are a concern in the cyber era. Attacks can be malicious activities. One of them is known as ARP poisoning attack activity, which attacks by falsifying a computer's identity through illegal access to retrieve confidential information in a target computer. Besides, it has also caused service deadlocks in the network. Previous studies have been introduced for the ARP Attack Detection model using rule-based and mining-based. Still, they cannot show optimal detection performance and obtain high false positive results. This paper proposed a detection model for ARP poisoning attacks using a similarity measurement approach adopting cosine similarity. The goal is to obtain measurements of host activities similar to ARP poisoning attacks. The experiment results showed that the model got an accuracy of 97.25%, recall of 96.43%, and precision of 81% with a similarity threshold value of 0.488. Comparison results with previous studies showed higher detection accuracy than previous studies and some classification methods. It shows that the model can improve intrusion detection performance and facilitate network administrators to analyze ARP poisoning attacks in computer networks.

Keywords: *ARP Poisoning, Similarity, Machine Learning, Network Security, Network Infrastructure*

1. Introduction

Nowadays, system security in computer networks needs to be handled appropriately. It also needs to avoid and detect malicious activities that can cause worse damage to computer networks and credential data. This malicious activity is often referred to as intrusion [1]. In dealing with intrusion in computer networks, an Intrusion detection model known as IDS [2] needs to be implemented in computer networks to strengthen communication services between interconnected computers.

Generally, IDS systems are built with two approaches, namely misuse detection and anomaly detection [3]. These approaches can be implemented based on rule-based formation, such as the SNORT application [4]–[7]. However, this IDS model cannot detect new types of attack variants in the network, and the detection accuracy depends on the accuracy of the rule base formation.

One type of attack with a high detection error in intrusion detection systems is the Address Resolution Protocol (ARP) Poisoning attack. ARP is a communications protocol that maps the addressing of each computing device in the computer network as a Media Access Control (MAC) address. Research on ARP attack detection has been introduced in previous research, such as detection models by applying the concept of mining-based analysis [1]–[3], [8], [9], anomaly behavior analysis [10]–[13] and used supporting applications [4], [14], [15]. In [16], an ARP poisoning detection model is introduced by separating ARP service access and using a listing technique in the form of static MAC address addressing. The experimental results show the model can detect suspected ARP Poisoning attack activity. Still, the model is very dependent on resources in the ARP control flow model. Selvarajan et al. [5] proposed an ARP poisoning detection model involving communication analysis on manipulated ICMP echo requests. The detection process involves rule-based analysis by checking communications on the ICMP protocol on ping messages that lead to ARP spoofing activity. The test results show that the model can detect behavior using a statistical approach based on the behavioral characteristics of suspected ARP poisoning perpetrators who communicate using certain protocol services. However, this research requires an attack scenario using a specific protocol. This type of ARP poisoning attack generally involves several communication protocols. [13] proposed an ARP attack detection model by distinguishing detection and prevention mechanisms. ARP attacks are divided into two activity processes: ARP spoofing and ARP poisoning attacks. The activity detection mechanism creates a static list of communications in the network, namely a list of IP address mapping and MAC address mapping tables. Then, the legal access in the permission list gets communication registration on the network. This model produces the detection and prevention of ARP poisoning attacks but requires a validation process on the network, which can hinder the duration of activity on the network. Besides, the model introduced uses statistical rules implemented for several hosts declared to be registered. In some cases, new hosts that have not yet been registered require a legal communication process.

The ARP poisoning model often uses analysis of network traffic data obtained from recordings [15] or uses public datasets that have been processed in the form of netflows [8], [10]. Besides, the Netflows are commonly used to detect attacks or malicious activity anomalies [17], including ARP Poisoning attacks. ARP poisoning is often misused to cause computer network deadlocks by illegally forging computer addresses to obtain confidential and credential information [8]. Thus, the proper ARP poisoning attack detection technique is required to accurately detect the attack in a computer network.

This paper proposed a new approach for ARP poisoning attack detection using a similarity analysis based on NetFlow analysis. This research is a development in [8], where the detection model used a classification approach in previous studies. The results of ARP poisoning attack detection using classification have a high detection accuracy but a high detection error value. Therefore, in the proposed research, the detection model aims to improve detection accuracy by suppressing detection errors based on NetFlow analysis. The novelty of this research is that it involves analyzing the dynamic similarity threshold value. In addition, this research built a knowledge base based on the characteristics of attack patterns. The similarity measurement results in the detection model are expected to show the closeness between the activity patterns of suspected attackers and the characteristics of the ARP poisoning attack knowledge base.

This paper is constructed into several sections. The process stages of the proposed model are introduced in Section II. Section III presents the results of the experiment and the research discussion. Finally, the conclusions of the research are drawn in section IV.

2. Research Methods

Research on ARP poisoning attack detection has been conducted by previous researchers. Some of them applied the concept of mining-based analysis [1]–[3], [8], [9], anomaly behavior analysis [10]–[13] and used supporting applications [4], [14], [15]. ARP poisoning activities often used analysis on network traffic data obtained from recording results [15] or used public datasets that had been processed in the form of network traffic flows (netflows) [8], [10]. The proposed ARP Poisoning detection model is shown in Figure 1.

p-ISSN 2088-1541
e-ISSN 2541-5832

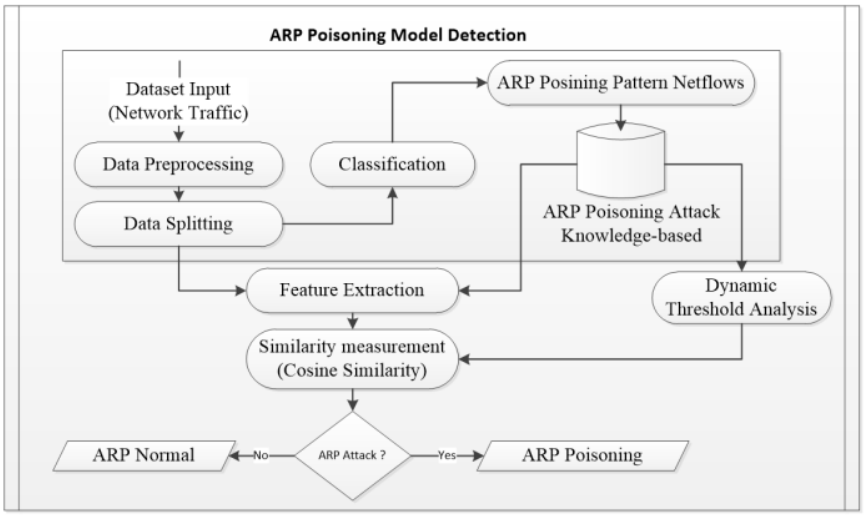

Figure 1. The proposed model

2.1. Dataset

This research used network traffic data obtained from the computer network recording process. The recorded data was in the form of .pcap files and processed into .csv files to form a dataset by adopting the techniques in research [8] and standardized based on the IDMEF standard [18]. Netflow data had a recording duration of 1 hour with 2819 traffic records, 279 ARP poisoning attack records, and 2540 normal activity traffic records. The number of attackers was six out of 418 hosts. ARP poisoning attacks involved malware to do ARP broadcasts, ARP flooding packets, and MAC flooding. Normal activities were conducted by hosts with browsing, e-mail sending, DNS access, and FTP access.

2.2. Data Preprocessing

At this phase, the data in .csv format are processed in the preprocessing process: data cleansing, normalization, and feature selection. The data cleansing stage was conducted to clean some data records that do not have values and are double data. Double data can appear due to errors in the recording process. In addition, the data cleansing process was carried out to fill in the "0" value to the feature value with a null value, aiming to standardize the value of each attribute in the data record. After the cleansing process, the normalized data were normalized using a value range of 0 to one, adopting the approach in research [19].

2.3. Data Splitting

Data splitting is dividing data into two types: training data and testing data. This stage was often used for the data learning process of a machine learning-based detection model [20]–[23] with a composition of 70% and 30% as testing data. In the testing data, the attack identities label of each record is removed and used to test the learning model used.

2.4. Classification

The classification stage is the ARP attack activity detection stage from the learning process. The detection results would record the attack activity and be stored in the knowledge base. The classification stage used five types of classification models, namely k-NN, Logistic Regression, Naïve Bayes, Random Forest, and Decision Tree. Evaluation of the five models with the best detection accuracy, precision, and recall values stored would be labeled as ARP poisoning pattern NetFlow.

2.5. ARP Poisoning Pattern NetFlow

ARP poisoning pattern flows are the knowledge base of ARP attack activities. Each data record would be labeled as an activity and sorted based on the time of the ARP attack activity. Thus, in the knowledge base of sequential activities among ARP attack activities.

2.6. Dynamic Threshold Analysis

Dynamic threshold analysis is a stage to determine the relationship between ARP attack activities. The intended relationship was the similarity between the two attacker activities. If two different attackers are represented as nodes A and B, there may be a difference in similarity between A to B and B to A [24]. To determine whether two attacking objects are similar and have a substantial similarity value, it is necessary to analyze the similarity threshold value [25]. In this paper, the similarity measurement (τ_{sim}) between two ARP attack patterns adopted the cosine similarity approach. The determined threshold value was dynamic based on the characteristics of the data using the (1):

$$T_{sim} = \frac{Max_{sim}+Min_{sim}}{2}, \quad (1)$$

where Min_{sim} Is the lowest similarity value that occurs from all the attacker activity similarity measurements, Max_{sim} is the highest value of the measured similarity value. The threshold value obtained will be updated if new attack characteristics are in the knowledge base update, namely in the ARP poisoning pattern NetFlow.

2.7. Feature Extraction

The Feature Extraction stage is the feature extraction of each data attribute or primary feature in NetFlow traffic. The number of primary attributes used in the previous process was ten features: *source port, source IP address, destination port, destination IP address*, *length, UDP port, TCP port*, *source MAC address, protocol*, *and destination MAC address*. If traffic in a computer network (N_T) consists of host activities in which there are traffic records (r_t) defined as $R_t = \{r_{t_1}, r_{t_2}, \dots, r_{t_j}\}$, containing feature *tuples* (F) in each record, namely *source IP address* (F_{SIP}), *destination IP address* (F_{DIP}), *protocol*(F_{Proto}), *TCP port* (F_{TPort}), *UDP port*(F_{UPort}), *length*(F_{length}), *source port*(F_{SPT}), *source MAC address*(F_{MSrc}) and *destination MAC address* (F_{Mdst}), thus denoted $n_T \in N_T$, where $n_T = \{(F_{SIP}, F_{DIP}, F_{TPort}, F_{UPort}, F_{length}, F_{SPT}, F_{Proto}, F_{MSrc}, F_{Mdst},)\}$.

Each feature tuple (F) value was re-extracted to get six new features denoted as FT by calculating the activity type of each host's interaction. The type of host activities can be in the form of spreading activity and show their pattern for communication behavior in the network. The type or variant of the feature is calculated by grouping each feature based on (F_{SIP}). Thus, each feature is defined as$FT_1 = n(F_{TPort}), FT_2 = n(F_{UPort}), FT_3 = n(F_{length}), FT_4 = n(F_{SPT}), FT_5 = n(F_{DIP}), FT_6 = n(F_{Proto})$.

The feature extraction results produced a feature pattern denoted as φ with a value of $\varphi = \{FT_1, FT_2, FT_3, FT_4, FT_5, FT_6,\}$. The feature pattern ($\varphi$) obtained from the classification results will be stored in the ARP poisoning attack activity knowledge base denoted by φ_{kb}.

2.8. Similarity Measurement

This stage measured the similarity between ARP attack traffic in the knowledge base as training data and network traffic as testing data. The similarity measurement adopted the cosine similarity shown in (2).

$$\text{sim}_{\cos}(x,y) = \cos(x,y) = \frac{x.y}{|x||y|}, \quad (2)$$

where the inner product is symbolized by the sign "." with the calculation:

$$x.y = \sum_{i=1}^{r} x_i y_i \quad (3)$$

and $|x|$ the calculation result of vector x :

$$|x| = \sqrt{\sum_{i=1}^{r} x_i^2} \quad (4)$$

and $|y|$ the calculation result of vector y :

$$|y| = \sqrt{\sum_{i=1}^{r} y_i^2} \quad (5)$$

If the traffic testing data is denoted as dt_ts, then the feature tuple extraction on the testing data becomes $F - dt_ts$, with the features $(FT - dt_ts)$ formed into a feature pattern denoted as φ_{dt_ts}. Thus, the similarity measurement to determine how substantial the similarity is with the cosine similarity between the ARP poisoning attack feature pattern in the knowledge base (φ_{kb}) and the traffic in the testing data (φ_{dt_ts}) become (6).

$$\text{sim}\omega_{(\varphi_{kb},\varphi_{dt_ts})} = \frac{\sum_{i=1}^{r} \varphi kb_i \varphi dt_ts_i}{|\sqrt{\sum_{i=1}^{r} \varphi kb_i^2}||\sqrt{\sum_{i=1}^{r} \varphi dt_ts_i^2}|}, \quad (6)$$

The similarity measurement results $sim\omega_{(\varphi_{kb},\varphi_{dt_ts})}$ Which has a value above τ_{sim} will state φ_{dt_ts} as ARP poisoning attack activity. Thus, it is expressed in (7):

$$sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} = \begin{cases} \text{state ARP Poisoning Attack if } sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} \geq \tau_{sim} \\ \text{state Normal aktivity if } sim\omega_{(\varphi_{kb},\varphi_{dt_ts})} < \tau_{sim} \end{cases}, \quad (7)$$

2.9. Evaluation

In this stage, the evaluation uses F-measure by measuring the precision, recall, and detection accuracy. To calculate the True Positive, True Negative, False Negative, and False Positive values are traced. The true Negative (TN) value is the number of regular activities detected as normal activities. False Positive (FP) is a normal activity but detected as an ARP poisoning attack activity. Meanwhile, True Positive (TP) is an ARP poisoning attack activity that is correctly detected as an ARP poisoning attack activity. False Negative (FN) is the opposite of True Positive, so ARP poisoning attack activity is detected as normal activity.

3. Result and Discussion

This research used the ARP poisoning attack dataset used in the study [8]. Network traffic data was taken through network traffic recording with the assistance of the Wireshark application [26] and produced files in the form of .pcap. The recording was carried out for 1 hour. This research used a computer with Intel Core i5-9300H processor specifications, 16 GB RAM, and 500 GB SSD storage capacity during the experiment and traffic data collection on the network.

3.1. Experiment

The recording data produced by the application was the .pcap extension. This data was processed into a comma-separated value (.csv) file. It aimed to convert traffic data from unstructured to structured tabular data by separating each column with a ";" separator. The conversion was done using the command line-based Tshark application, as shown in Figure 2.

p-ISSN 2088-1541
e-ISSN 2541-5832

```
root@ubuntu:/home/sysadmin/arp# tshark -r TCPdump_Inside_Labeling.pcap -T fields -e frame.time -e ip
.src -e eth.src -e tcp.port -e udp.port -e frame.len -e ip.proto -e ip.dst -e eth.dst -E separator=,
 -E occurrence=f
```

Figure 2. Tshark command

After that, preprocessing was carried out, namely data cleansing, data normalization, and feature selection. In the data cleansing stage, 153 traffic records were deleted because they were redundant data. Redundant data could occur during the recording to conversion process using two applications, Wireshark and Tshark. In addition to removing redundant data, the null value in each column was filled to 0. There were 201 record data that had columns with null values and converted to a 0 value. The data cleansing process resulted in a traffic reduction of 5.43%. The details of the cleansing results are shown in Table 1.

Table 1. Preprocessing Result

Data before preprocessing	Data after preprocessing	Reduction Percentage
2,819	2,666	5.43 %

The results of data cleansing were followed by converting the values in each category data column into numerical values. This change technique used an encoder, also done in research [27]. Furthermore, all values in each column were normalized to a value scale between 0 and 1. Each normalized column was expressed as a traffic data feature. The basic features obtained with the Tshark application in Figure 2 of 7 features. In this research, six features were selected, and the selection was conducted manually. One feature that was not used was the time feature. The time feature was not used because it did not match the characteristics of ARP poisoning attacks, where attack activities occurred randomly and not continuously. ARP attacks tend to be characterized by spreading and have high-intensity occurrences. Features used were *source IP* (F_{SIP}), *destination IP* (F_{DIP}), *TCP port* (F_{TPort}), *UDP port*(F_{UPort}), *length*(F_{length}), *source port*(F_{SPT}), *protocol*(F_{Proto}), *source MAC address*(F_{MSrc}) and *destination MAC address* (F_{Mdst}).

From the preprocessing results, the data was divided into two types of data, namely training data as the knowledge base of ARP attacks (φ_{kb}) after the classification stage and testing data as denoted by (φ_{dt_ts}). Data were divided randomly with data divided composition shown in Table 2.

Table 2. Data Composition

Traffic Record		Data Training (70%)		Data Testing (30%)	
ARP attack	Normal	ARP attack	Normal	ARP attack	Normal
279	2,387	195	1,671	84	716
Total Traffic after Preprocessing: 2,666		Total Traffic: 1,866		Total Traffic: 800	

In this research, the classification was performed using five classification methods, namely k-NN, Logistic Regression, Naïve Bayes, Random Forest, and Decision Tree. The classification results are shown in Figure 3.

p-ISSN 2088-1541
e-ISSN 2541-5832

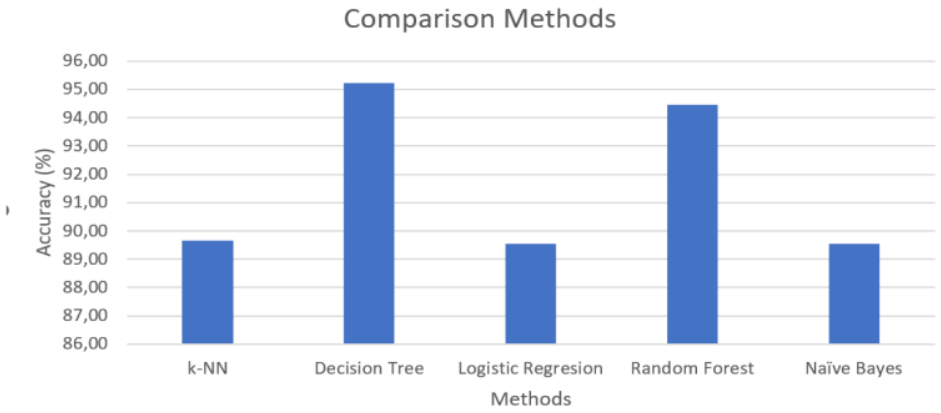

Figure 3. Evaluation results of 5 classification methods

The method with the best detection accuracy will be used to build the ARP poisoning attack knowledge base. The results of the five classification methods are shown in Table 3.

Table 3. Classification result

No.	Classification method	Accuracy
1	k-NN	89.65
2	Random Forest	94.45
3	Naïve Bayes	89.53
4	Logistic Regression	89.53
5	Decision Tree	95.24

The classification results showed that the decision tree method obtained the highest detection accuracy by producing 196 traffic records as ARP poisoning attack patterns. Furthermore, the traffic records were extracted into FT, where the variants of each feature were grouped based on (F_{SIP}) and produced a total of $n = (FT_i)$ with i = 6 namely $F_{TPort}, F_{UPort}, F_{length}, F_{SPT}, F_{DIP}, F_{Proto}$. The feature extraction results formed feature patterns $\varphi = \{FT_1, FT_2, FT_3, FT_4, FT_5, FT_6,\}$ with the value of each FT is $FT_1 = n(F_{TPort}), FT_2 = n(F_{UPort}), FT_3 = n(F_{length}), FT_4 = n(F_{SPT}), FT_5 = n(F_{DIP}), FT_6 = n(F_{Proto})$. Examples of the knowledge base (φ_{kb}) of ARP poisoning attack patterns is shown in Table 4.

Table 4. Traffic records on φ_{kb}

φ_{kb_i}	$n(F_{TPort})$	$n(F_{UPort})$	$n(F_{length})$	$n(F_{SPT})$	$n(F_{DIP})$	$n(F_{Proto})$	ARP Poisoning Attack
1	0.7727	0.2857	0.8421	0.3333	0.4795	0.9091	1
2	0.1364	0.4762	0.9474	0.8182	0.6438	0.9091	1
3	0.6364	0.7143	0.8947	0.3939	0.3562	0.9091	1
4	0.6136	0.1429	0.1053	0.3030	0.7808	0.9091	1
5	0.8636	0.1905	0.5789	0.9697	0.6712	0.9091	1
6	0.4318	0.8571	0.1053	0.5455	0.0548	0.9091	1
7	0.4318	0.6190	0.1579	1,0000	0.0959	0.9091	1
8	0.3864	0.3810	0.3684	0.7879	0.0685	0.9091	1
9	0.4318	0.4286	0.1579	0.6970	0.9589	0.9091	1
10	0.6136	0.2857	0.3684	0.6970	0.9863	0.9091	1
11	0.4091	0.3810	0.5789	0.7879	0.4521	0.9091	1
12	0.9545	0.3333	0.0526	0.2424	0.2329	0.9091	1
13	0.7500	0.4286	0.1579	0.3030	0.6301	0.9091	1
14	0.0455	0.5714	0.8947	0.5455	0.0137	0.8182	1
15	0.5455	0.8571	0.7895	0.0303	0.3836	0.8182	1
…	…	…	…	…	…	…	…

p-ISSN 2088-1541
e-ISSN 2541-5832

In the similarity measurement, the testing data used was 30% of the dataset divided at the data splitting stage. Examples of testing data that have been formed into feature patterns φ_{dt_ts} are shown in Table 5.

Table 5. Data Testing

$\varphi_{dt_ts_i}$	$n(F_{TPort})$	$n(F_{UPort})$	$n(F_{length})$	$n(F_{SPT})$	$n(F_{DIP})$	$n(F_{Proto})$	ARP Poisoning Attack
1	0.2045	0.1429	0.0526	0.5455	0.0137	0.2727	?
2	0.2045	0.2381	0.0000	0.6364	0.5890	0.8182	?
3	0.5682	0.7619	0.1579	0.8788	0.6301	0.8182	?
4	0.3864	0.1905	0.8947	0.3030	0.6712	0.2727	?
5	0.2727	0.3810	1,0000	0.6364	0.1096	0.0909	?
6	0.4545	0.6667	0.7895	0.0606	0.7808	0.4545	?
7	0.2045	0.2857	0.7368	0.8788	0.1781	0.8182	?
8	0.4318	0.6667	0.6842	1,0000	0.6301	0.5455	?
9	0.2727	0.4286	0.5789	0.6970	0.8082	0.5455	?
10	0.3409	0.9524	0.4737	0.7576	0.4658	0.2727	?
11	0.9545	0.6190	1,0000	0.2121	0.5205	0.3636	?
…	…	…	…	…	…	…	…

The results of similarity measurement between the ARP poisoning attack feature pattern in the knowledge base (φ_{kb}) and the traffic in the testing data (φ_{dt_ts}) result in several similarity values. Each data record in φ_{dt_ts} will be measured against 196 data records in φ_{kb}. The value taken as the similarity result was the average value of the total similarity with the φ_{kb}. Data record. The results of similarity measurement are shown in Table 6.

Table 6. Example of Cosine Matrix Map Results

		φ_{dt_ts}										
		1	2	3	4	5	6	7	8	9	10	11
φ_k	1		0.197	0.141	0.519	0.368	0.585	0.146	0.165	0.270	0.246	…
	2	0.197		0.075	0.400	0.563	0.371	0.200	0.175	0.132	0.288	…
	3	0.141	0.075		0.327	0.388	0.249	0.176	0.071	0.097	0.100	…
	4	0.519	0.400	0.327		0.160	0.086	0.243	0.158	0.109	0.265	…
	5	0.368	0.563	0.388	0.160		0.282	0.171	0.155	0.235	0.202	…
	6	0.585	0.371	0.249	0.086	0.282		0.325	0.181	0.133	0.193	…
	7	0.146	0.200	0.176	0.243	0.171	0.325		0.092	0.133	0.218	…
	8	0.165	0.175	0.071	0.158	0.155	0.181	0.092		0.033	0.051	…
	9	0.270	0.132	0.097	0.109	0.235	0.133	0.133	0.033		0.117	…
	10	0.246	0.288	0.100	0.265	0.202	0.193	0.218	0.051	0.117		…
	…	…	…	…	…	…	…	…	…	…	…	…

From the similarity measurement results, the lowest value of $sim\omega_{(\varphi_{kb},\varphi_{dt_ts})}$ was 0.033, and the highest value was 0.862. Thus, the value of τ_{sim} obtained was 0.448 as the threshold value for determining ARP poisoning traffic. In this research, the value of τ_{sim} = 0.448 successfully obtained six hosts of ARP Poisoning attack perpetrators based on the *source IP* (F_{SIP}) grouping with the identification accuracy of the number of ARP poisoning attack traffic 82 and normal activity of 709 records. Identification details of ARP poisoning attacks based on similarity measurements are shown in Table 7.

Table 7. Detailed evaluation results of ARP Attack detection

TP	FP	FN	TN	Accuracy	Precision	Recall
82	7	2	709	98.88	92.13	97.62

p-ISSN 2088-1541
e-ISSN 2541-5832

The similarity measurement results successfully detect ARP Poisoning traffic records with a detection accuracy of 98.88%, precision of 92.13%, and recall of 97.62%. It showed that the proposed model performs optimally to detect ARP poisoning attacks.

3.2. Analysis and Discussion

This research proposed a new ARP Poisoning attack detection model with a similarity measurement approach and dynamic threshold value analysis. In the preprocessing stage, there was a traffic reduction of 5.43%, and only normal activity occurred. Some normal activities had the potential to be performed repeatedly by the user, thus causing recording as redundant data by the Wireshark application and Tshark application conversion.

In analyzing threshold value, in addition to using (1), this research also conducted a heuristic analysis to obtain the optimal τ_{sim} value analysis. The result of the similarity threshold value search found that the best τ_{sim} value was 0.488, with the highest accuracy, precision, and recall values. Thus, it was determined that τ_{sim} =0.488 was the optimal threshold value used to detect ARP Poisoning attacks on traffic records in the testing data. The threshold value can dynamically change if there is a change in the results of the knowledge base formation that depends on the classification method. The higher the accuracy value of the classification method, the more optimal the knowledge base formation in the form of ARP attack feature patterns. The optimization of knowledge base formation shows a more significant amount of feature pattern data. The results of the threshold value search are shown in Figure 4.

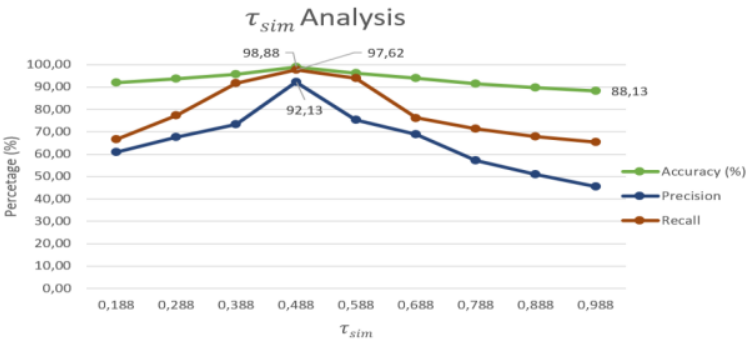

Figure 4. Optimal value tracing results from τ_{sim}

In this paper, the proposed model could detect ARP Poisoning attacks with a detection accuracy value of 97.25%, precision of 81%, and recall of 96.43%. These results had higher values than research [8] and detection results in several classification methods. The comparison results are shown in Table 8.

Table 8. Comparison with previous studies

Methods and Previous Studies	Accuracy	Precision	Recall	Threshold analysis
Atmojo et al. [8] k-NN (k=2)	98,7	98,7	98,6	No
Random Forest	94,5	95,1	99,2	No
Linear Regression	91,9	92,1	99,6	No
Naïve Bayes	96,7	97,7	99	No
Support Vector Machine	95,6	96,8	98,8	No
Decision Tree	97,5	99,0	98,5	No
Proposed Model	**98.88**	**92.13**	**97.62**	**yes**

The comparison results with previous research; the proposed model had higher detection accuracy than earlier research, which was 98.88%. The detection accuracy that could be obtained in this research was 0.1% higher than in the previous study. However, the precision and recall values had lower values. The precision value was 6.87% lower than the highest precision value produced by the Decision Tree method. At the same time, the recall value was 1.92% lower than the highest recall value produced by the Linear Regression classification method. The precision and recall values were obtained lower because the composition of the number of ARP attack activities was only 10.5% of the total traffic data in the testing data and imbalanced compared to the number of normal activity data records. Besides, this paper has the novelty of a detection process that involves the analysis of dynamic similarity threshold values that have never been used in previous research.

4. Conclusion

This paper proposed a detection model to detect ARP poisoning attacks using a similarity analysis approach adopting cosine similarity. The proposed model aimed to obtain substantial similarity between host activities the network and ARP Poisoning attack activities in the knowledge base formed from the machine learning-based classification model. The proposed model had a novelty in analysis that involved a dynamic threshold value to determine the host activity pattern as an ARP Poisoning attack. In this research, the model successfully detected ARP Poisoning attacks with a detection accuracy value of 97.25%, precision of 81%, and recall of 96.43% with a similarity threshold value of 0.488. Detection accuracy showed higher results than Atmojo et al. [8] and some machine learning-based classification methods. The detection accuracy that could be obtained in this research was 0.1% higher than the highest accuracy value in previous research obtained by the k-NN classification method with a value of $k = 2$. However, the recall and precision had lower values. The precision value was 6.87% lower than the highest value the Decision Tree method produced. At the same time, the recall value was 1.92% lower than the highest recall value produced by the Linear Regression classification method. These two lower precision and recall values were caused by the data composition formed during network traffic recording. However, the composition of the number of traffic records recorded with the Wireshark application was a characteristic that corresponded to the actual occurrence of ARP Poisoning attacks on the network. In this paper, the proposed model could be used to develop intrusion detection models and make it easier for network administrators to analyze ARP poisoning attacks in computer networks.

The model needs to be developed in future research by optimizing features and feature selection. It aims to improve the model evaluation of the precision and recall measurements without degrading the current detection accuracy results. The precision and recall values need to be increased in further research by handling the imbalance data issues to reduce the positive error rate. In addition, it can develop a time of occurrence analysis to obtain attack analysis that can be causal attacks. Thus, the ARP Poisoning attack detection model could be performed optimally.

References

[1] G. Pu, L. Wang, J. Shen, and F. Dong, "A hybrid unsupervised clustering-based anomaly detection method," *Tsinghua Tsinghua Science and Technology*, vol. 26, no. 2, pp. 146–153, 2021, doi: 10.26599/TST.2019.9010051.

[2] V. Bhatia, S. Choudhary, and K. R. Ramkumar, "A Comparative Study on Various Intrusion Detection Techniques Using Machine Learning and Neural Network," *ICRITO 2020 - IEEE 8th Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, pp. 232–236, 2020, doi: 10.1109/ICRITO48877.2020.9198008.

[3] L. Mohan, S. Jain, P. Suyal, and A. Kumar, "Data mining Classification Techniques for Intrusion Detection System," *Proceedings - 2020 12th International Conference on Computational Intelligence and Communication Networks, CICN 2020*, pp. 351–355, 2020, doi: 10.1109/CICN49253.2020.9242642.

[4] A. Tasneem, A. Kumar, and S. Sharma, "Intrusion Detection Prevention System using SNORT," *International Journal of Computer Applications*, vol. 181, no. 32, pp. 21–24, 2018, doi: 10.5120/ijca2018918280.

[5] S. Selvarajan, M. Mohan, and B. R. Chandavarkar, "Techniques to Secure Address Resolution Protocol," *2020 11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020*, doi: 10.1109/ICCCNT49239.2020.9225413.
[6] Z. Trabelsi and W. El-Hajj, "ARP spoofing: A comparative study for education purposes," *Proceedings of the 2009 Information Security Curriculum Development Annual Conference, InfoSecCD'09*, pp. 60–66, 2009, doi: 10.1145/1940976.1940989.
[7] V. Srivastava and D. Singh, "Enhance detecting and preventing scheme for ARP Poisoning using DHCP," Computer Modelling and New Technologies, vol. 21, no. 2, pp. 93–99, 2017.
[8] Y. P. Atmojo, I. M. D. Susila, I. B. Suradarma, L. Yuningsih, E. S. Rini, and D. P. Hostiadi, "A New Approach for ARP Poisoning Attack Detection Based on Network Traffic Analysis," *2021 4th International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2021*, pp. 18–23, 2021, doi: 10.1109/ISRITI54043.2021.9702860.
[9] A. Krishna, M. A. Ashik Lal, A. J. Mathewkutty, D. S. Jacob, and M. Hari, "Intrusion Detection and Prevention System Using Deep Learning," *Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020*, pp. 273–278, 2020, doi: 10.1109/ICESC48915.2020.9155711.
[10] V. Rohatgi and S. Goyal, "A detailed survey for detection and mitigation techniques against ARP spoofing," Proceedings of the 4th International Conference on IoT in Social, Mobile, Analytics, and Cloud, ISMAC 2020, pp. 352–356, 2020, doi: 10.1109/I-SMAC49090.2020.9243604.
[11] M. Ren, Y. Tian, S. Kong, D. Zhou, and D. Li, "A detection algorithm for ARP man-in-the-middle attack based on data packet forwarding behavior characteristics," *Proceedings of 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference, ITOEC 2020*, pp. 1599–1604, 2020, doi: 10.1109/ITOEC49072.2020.9141555.
[12] S. Sun, X. Fu, B. Luo, and X. Du, "Detecting and mitigating ARP attacks in SDN-based cloud environment," *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops 2020*, pp. 659–664, 2020, doi: 10.1109/INFOCOMWKSHPS50562.2020.9162965.
[13] S. Hijazi and M. S. Obaidat, "A New Detection and Prevention System for ARP Attacks Using Static Entry," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2732–2738, 2019, doi: 10.1109/JSYST.2018.2880229.
[14] S. Ahn, T. Lee, and K. Kim, "A Study on Improving Security of ICS through Honeypot and ARP Spoofing," *ICTC 2019 - 10th International Conference on ICT Convergence: ICT Convergence Leading the Autonomous Future*, pp. 964–967, 2019, doi: 10.1109/ICTC46691.2019.8939925.
[15] M. Abid and A. Singh, "Arp Spoofing Detection via Wireshark and Veracode," *International Journal of New Technology and Research*, vol. 4, no. 5, p. 263063, 2018.
[16] H. Y. Ibrahim, P. M. Ismael, A. A. Albabawat, and A. B. Al-Khalil, "A Secure Mechanism to Prevent ARP Spoofing and ARP Broadcasting in SDN," *Proceedings of the 2020 International Conference on Computer Science and Software Engineering, CSASE 2020*, pp. 13–19, 2020, doi: 10.1109/CSASE48920.2020.9142092.
[17] T. Yu and R. Yue, "Detecting Abnormal Interactions among Intranet Groups Based on Netflow Data," *IOP Conference Series: Earth and Environmental Science*, vol. 428, no. 1, 2020, doi: 10.1088/1755-1315/428/1/012039.
[18] H. Debar, "The IDMEF : RFC 4765," *Mycological Research*, 2007.
[19] D. P. Hostiadi and T. Ahmad, "Hybrid model for bot group activity detection using similarity and correlation approaches based on network traffic flows analysis," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 7, pp. 4219–4232, 2022, doi: 10.1016/j.jksuci.2022.05.004.
[20] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *Journal Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-020-00390-x.
[21] M. N. Aziz and T. Ahmad, "Clustering under-sampling data for improving the performance of intrusion detection system," *Journal of Engineering Science and Technology*, vol. 16, no. 2, pp. 1342–1355, 2021.
[22] F. Selahshoor, H. Jazayeriy, and H. Omranpour, "Intrusion Detection systems using Real-Valued Negative Selection Algorithm with Optimized Detectors," *5th Iranian Conference on Signal Processing and Intelligent Systems, ICSPIS 2019*, pp. 18–19, 2019, doi: 10.1109/ICSPIS48872.2019.9066040.

[23] P.-C. Chang, Y.-W. Wang, and C.-H. Liu, "The development of a weighted evolving fuzzy neural network for PCB sales forecasting," *Expert Systems with Applications*, vol. 32, no. 1, pp. 86–96, 2007, doi: https://doi.org/10.1016/j.eswa.2005.11.021.

[24] D. P. Hostiadi, W. Wibisono, and T. Ahmad, "B-Corr Model for Bot Group Activity Detection Based on Network Flows Traffic Analysis," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 10, pp. 4176–4197, 2020, doi: 10.3837/tiis.2020.10.014.

[25] Y. Zou, F. Dong, B. Lei, S. Sun, T. Jiang, and P. Chen, "Maximum similarity thresholding," *Digital Signal Processing*, vol. 28, pp. 120–135, 2014, doi: 10.1016/j.dsp.2014.02.008.

[26] J. C. Vega, M. A. Merlini, and P. Chow, "FFShark: A 100G FPGA Implementation of BPF Filtering for Wireshark," *Proceedings - 28th IEEE International Symposium on Field-Programmable Custom Computing Machines, FCCM 2020*, pp. 47–55, 2020, doi: 10.1109/FCCM48280.2020.00016.

[27] M. A. R. Putra, T. Ahmad, and D. P. Hostiadi, "Analysis of Botnet Attack Communication Pattern Behavior on Computer Networks," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 4, pp. 533–544, 2022, doi: 10.22266/ijies2022.0831.48.

The Optimization of the Poisoning Attack Detection Model Using a Similar Approach Based on NetFlow Analysis

ORIGINALITY REPORT

7%
SIMILARITY INDEX

PRIMARY SOURCES

1	ojs.unud.ac.id Internet	170 words — 3%
2	Dandy Pramana Hostiadi, Yohanes Priyo Atmojo, Roy Rudolf Huizen, I Made Darma Susila, Gede Angga Pradipta, I Made Liandana. "A New Approach Feature Selection for Intrusion Detection System Using Correlation Analysis", 2022 4th International Conference on Cybernetics and Intelligent System (ICORIS), 2022 Crossref	120 words — 2%
3	Dandy Pramana Hostiadi, Yohanes Priyo Atmojo, Roy Rudolf Huizen, I Made Darma Susila, Gede Angga Pradipta, I Made Liandana. "Correlation-Based Feature Selection on Botnet Activity Detection Using Kendall Correlation", 2022 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), 2022 Crossref	85 words — 2%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	< 2%
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF